PERSONAL COMPUTER SYSTEM

Patent number: JP3091838

Publication date: 1991-04-17

Inventor: RICHIYAADO BIIIRUKOFUSUKII; JIYON UIREI BURATSUKURETSUJI J; DOIRU SUTANFUIIRU KURONKU; RICHIYAADO AREN DAIAN; SUKOTSUTO JIERARUDO KINIAA; JIYOOJI DEII KOBATSUKU; MASHIYUU SUTEIBUN PORUKA JIYUN; ROBAATO SAKUSENMAIAA; KEBIN MAASHIYARU JIBOROSUKII; JIERII DEYUUN DEIKISHION; ANDORIYUU BOISU MAKUNEIRU; EDOWAADO IIBUNINGU WATSUCHITER

Applicant: IBM

Classification:

- international: G06F9/445; G06F13/00; G06F13/10

- european: G06F9/445B4

Application number: JP19900220132 19900823

Priority number(s): US19890398820 19890825

Also published as:

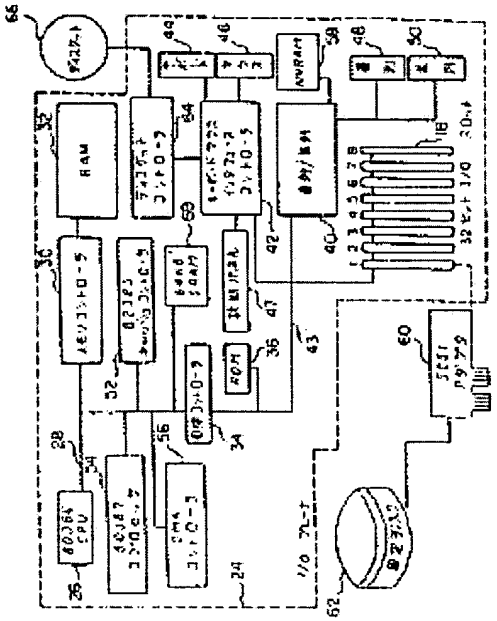
EP0417889 (A2)
US5022077 (A1)
MX171879 (A)
EP0417889 (A3)
DE4026912 (A1)

more >>

Report a data error here

Abstract of **JP3091838**

PURPOSE: To prevent a BIOS code stored in a fixed disk from being changed without permission by providing a system processor, ROM, RAM, direct access storage device, and the same controller. CONSTITUTION: The first part of a BIOS is stored in an ROM 36, a direct access storage device 32 stores a master boot record including an executable code segment and the second part of the BIOS in a specific area, and a memory controller 30 protects the specific area, and permits access in response to a reset signal. A processor 26 is initialized by the first part of the BIOS of the ROM 36, and the reset signal is applied to the controller 30, and a CPU 26 performs access to the master boot record in order to set the master boot record loadable to the ROM 32, and moves the control to the executable code segment. The second part of the BIOS is loaded to the RAM 32, the control is moved, and the OS is booted. The second part prevents the access to the specific area by the protecting means of the controller 30 during the normal operation of the OS. Thus, the unauthorized change of the BIOS of a fixed disk 62 with especially large capacity can be prevented.



⑲日本国特許庁(JP)　⑪特許出願公開

⑫公開特許公報(A)　平3-91838

⑤1Int.Cl.⁵	識別記号	庁内整理番号	④3公開　平成3年(1991)4月17日
G 06 F 9/445 13/00	305 J	7629-5B 7361-5B	G 06 F 9/06　420 H※

審査請求　有　請求項の数 24　(全22頁)

⑤4発明の名称　パーソナルコンピュータシステム

⑳特　願　平2-220132
㉒出　願　平2(1990)8月23日
優先権主張　㉜1989年8月25日㉝米国(US)㉛398820

⑦2発明者　リチャード・ビイールコフスキイ　アメリカ合衆国フロリダ州デルレイ・ビーチ、ハミングバード・ドライブ1401番地
⑦2発明者　ジョン・ウイレイ・ブラックレツジ、ジユニア　アメリカ合衆国フロリダ州ボカ・ラトン、シイクワイア・レーン304番地
⑦1出願人　インターナシヨナル・ビジネス・マシーンズ・コーポレーション　アメリカ合衆国10504、ニユーヨーク州　アーモンク(番地なし)
⑦4代理人　弁理士　頓宮　孝一　外1名
最終頁に続く

明　細　書

1．発明の名称　パーソナルコンピュータシステム

2．特許請求の範囲

(1) オペレーテイングシステムを実行するシステムプロセツサと、

ＢＩＯＳの第１部分を記憶する読み取り専用メモリと、

ランダムアクセスメモリと、

実行可能コードセグメントを含むマスタブートレコード及び前記ＢＩＯＳの第２部分を特定の領域に記憶する直接アクセス記憶装置と、

前記特定の領域を保護し、リセツト信号に応答して前記特定の領域へのアクセスを許可する保護手段を有する直接アクセス記憶装置コントローラとを備え、

前記第１部分はシステムプロセツサを初期設定するとともに、前記リセツト信号を発生して前記コントローラへ供給することにより、前記システムプロセツサが前記マスタブートレコードをアクセスして前記ランダムアクセスメモリにロードできるようにし、更に制御を前記実行可能コードセグメントに移し、

前記実行可能セグメントは制御が移されたことに応答して、前記第２部分を前記ランダムアクセスメモリにロードし、更に前記オペレーテイングシステムをブートするために制御を前記第２部分に移し、

前記第２部分は、前記オペレーテイングシステムの通常動作の間、前記特定の領域のアクセスを禁止するために前記保護手段を活動化する、

ことを特徴とするパーソナルコンピュータシステム。

(2) 前記第１部分は、電源投入により、前記リセツト信号の発生を開始する、請求項１に記載のシステム。

(3) 前記第１部分は、システムがリセツト状態になつたことに応答して、前記リセツト信号の発生を開始する、請求項１に記載のシステム。

(4) 前記マスタブートレコードは、該レコードに適合したハードウエア構成を表わすデータセグメントを含み、前記読み取り専用メモリは、前記システムプロセッサのハードウエア構成を表わすデータを含み、前記第１部分は、前記第２部分が前記ランダムアクセスメモリにロードされる前に、前記マスタブートレコード及び前記読み取り専用メモリからのハードウエア構成データを比較して、前記マスタブートレコードが前記システムプロセッサに適合しているかどうかを確かめる、請求項１に記載のシステム。

(5) 前記データセグメントは、前記マスタブートレコードに適合したシステムプレーナを表わす値を含み、システムプレーナは、自身を一意的に識別する手段を有する、請求項４に記載のシステム。

(6) 前記マスタブートレコードのハードウエア構成データは、該マスタブートレコードに適合したシステムプロセッサを識別するモデル値と、該マスタブートレコードに適合したシステムプレーナのＩ／Ｏ構成を表わすサブモデル値とを含み、前記読み取り専用メモリは、システムプロセッサを識別する対応モデル値及びシステムプレーナのＩ／Ｏ構成を表わす対応サブモデル値を含み、前記モデル値及び前記サブモデル値を前記対応モデル値及び前記対応サブモデル値とそれぞれ比較することにより、前記マスタブートレコードが前記システムプロセッサ及び前記システムプレーナのＩ／Ｏ構成に適合しているかどうかを確かめる、請求項４に記載にシステム。

(7) 前記システムプロセッサと電気的に接続された不揮発性ランダムアクセスメモリを更に備え、該メモリはシステム構成を表わすデータを含み、該データはシステムの構成が変わると更新され、前記第１部分は、該データを前記読み取り専用メモリ中の対応するデータと比較することにより、システムの構成が変わったかどうかを調べる、請求項１に記載のシステム。

(8) 前記システムプロセッサは、順番に番号を付けられたブロックの形で前記コントローラへデータレコードを転送し、前記マスタブートレコード及び前記第２部分は番号の大きい方のブロックにある、請求項１に記載のシステム。

(9) 前記保護手段はアドレス可能な最大ブロックを設定し、該最大ブロックは前記マスタブートレコード及び前記第２部分の最も番号の小さいブロックであり、前記保護手段は前記最大ブロック以上の番号のブロックへのアクセスを禁止するとともに、前記最大ブロックより小さい番号のブロックへのアクセスを許可する、請求項８に記載のシステム。

(10) システムプロセッサと、

主メモリと、

複数のデータレコード、該データレコードを前記主メモリへロードするためのロード手段、および該ロード手段によって前記主メモリに読み込まれる主メモリ常駐プログラムイメージを含み、そのうち前記ロード手段および前記主メモリ常駐プログラムイメージは保護可能な領域に記憶している少なくとも１つの直接アクセス記憶装置と、

前記システムプロセッサを初期設定するとともに、前記直接アクセス記憶装置から前記主メモリに前記ロード手段を読み込ませる第１プログラムを含む読み取り専用メモリと、

前記主メモリ常駐プログラムイメージが前記主メモリに読み込まれることによって生成される主メモリ常駐プログラムにより活動化され、前記直接アクセス記憶装置の前記保護可能な領域を保護して、前記ロード手段及び前記主メモリ常駐プログラムイメージが許可なくアクセスされるのを禁止する保護手段と、

を具備するパーソナルコンピュータシステム。

(11) 前記ロード手段は、当該システムが前記主メモリ常駐プログラムに適合しているかどうかを確かめる検査手段を含む、請求項１０に記載のシステム。

(12) 前記ロード手段は、前記主メモリ常駐プログラムのローディングを実行するための実行可能コードセグメントを有するマスタブートレコードを含み、前記第１プログラムが前記実行可能コードセグメントに制御を移すことによって前記主メモリ

への前記主メモリ常駐プログラムイメージのローディングを行わせる、請求項１１に記載のシステム。

(13)前記第１プログラムは、前記主メモリ常駐プログラムをロードするのに必要なシステム機能だけを初期設定して検査する電源投入自己検査ルーチンを含む、請求項１０に記載のシステム。

(14)前記電源投入自己検査ルーチンは、システムプロセッサ機能、メモリサブシステムおよび直接アクセス記憶装置サブシステムを初期設定する、請求項１３に記載のシステム。

(15)前記検査手段は、システムプロセッサのタイプおよびシステムプロセッサに接続されるシステムプレーナの構成を表わすデータを含み、請求項１１に記載のシステム。

(16)前記直接アクセス記憶装置は、固定ディスクおよびディスクコントローラを含み、前記システムプロセッサは番号を付されたブロックの形でデータを前記ディスクコントローラへ転送し、前記保護可能な領域には番号の大きい方のブロックが割当てられる、請求項１０に記載のシステム。

(17)前記保護手段は、前記システムプロセッサからのブロック番号と前記保護可能な領域に割当てられたブロックの最小番号を比較し、前記ブロック番号が該最小番号よりも小さい場合にのみアクセスを許可する、請求項１６に記載のシステム。

(18)前記第１プログラムは、電源投入またはリセット条件に応答してリセット信号の発生を開始する、請求項１０に記載のシステム。

(19)システムプロセッサと、

予め決められた最小番号から最大番号までの複数のブロックを記憶することができ、そのうち番号が第１の値以上のブロックにＢＩＯＳが記憶されている記憶装置と、

前記システムプロセッサと前記記憶装置の間に接続され、前記システムプロセッサからのブロック形式の入出力要求を前記記憶装置の物理的特性に合った形式の要求に変換する制御手段と、

リセット信号の発生を開始する第１論理手段と、

前記ＢＩＯＳへのアクセスを禁止する第２の信号を発生する第２論理手段と、

前記リセット信号に応答して前記ＢＩＯＳへのアクセスを許可し、前記第２の信号に応答して前記第１の値のブロックのところに境界を設定することにより前記ＢＩＯＳへのアクセスを禁止する保護手段と、

を具備するパーソナルコンピュータシステム。

(20)前記記憶装置はシリンダーヘッドーセクタ形式の要求を受付ける固定ディスクであり、前記制御手段は前記ブロック形式を前記シリンダーヘッドーセクタ形式に変換する、請求項１９に記載のシステム。

(21)前記第１論理手段は電源投入またはキーボード入力に応答して前記リセット信号の発生を開始する、請求項１９に記載のシステム。

(22)システムプロセッサ、システムを初期設定するための手段を含む第１ＢＩＯＳ部分を記憶する読取専用メモリ、ランダムアクセスメモリおよび直接アクセス記憶装置を含むパーソナルコンピュータシステムにおいてＢＩＯＳを保護するための方法であつて、

実行可能コードセグメントを含むマスタブートレコードと、前記パーソナルコンピュータシステムの通常動作中は前記ランダムアクセスメモリに常駐している第２ＢＩＯＳ部分とを前記直接アクセス記憶装置の保護可能な区画に記憶し、

システムを初期設定して、前記直接アクセス記憶装置に送られるリセット信号の発生を開始し、

前記リセット信号に応答して前記保護可能な区画の保護を解除することにより、前記システムプロセッサが前記マスタブートレコードおよび前記第２ＢＩＯＳ部分をアクセスできるようにし、

前記マスタブートレコードを前記ランダムアクセスメモリにロードし、

制御を前記実行可能コードセグメントに移して、前記第２ＢＩＯＳ部分を前記ランダムアクセスメモリにロードし、

制御を前記ランダムアクセスメモリにある第２ＢＩＯＳ部分に移して、前記保護可能な区画へのアクセスを禁止する、

ことを特徴とするＢＩＯＳ保護方法。

(23)前記第１ＢＩＯＳ部分に含まれるデータと、前記マスタブートレコードに含まれる対応するデータとを比較することにより、前記マスタブートレコードがシステムに適合しているかどうかを確かめる、請求項２２に記載の方法。

(24)前記読取専用メモリに含まれるデータと、前記マスタブートレコードに含まれる対応するデータとを比較することにより、前記マスタブートレコードが前記システムプロセッサに適合しているかどうかを確かめる、請求項２２に記載の方法。

３．発明の詳細な説明

Ａ．産業上の利用分野

本発明はパーソナルコンピュータシステム（以下、ＰＣＳという）に関し、特にその大容量記憶装置に記憶されているＢＩＯＳを保護するための方法及び装置に関する。

Ｂ．従来の技術

ＰＣＳは現代社会の様々な分野で広く使用されるようになってきた。ＰＣＳは一般にデスクトップ型、フロアスタンド型、ポータブル型等に分けられるが、いずれもシステムプロセッサ、ディスプレイモニタ、キーボード、１以上のディスケットドライブ、固定ディスク記憶装置、（オプションの）プリンタ等を備えている。このようなシステムの１つの特徴は、マザーボードあるいはシステムプレーナを用いてそれらの構成要素を電気的に接続していることである。ＰＣＳは主としてシングルユーザー向けに設計されており、個人あるいは中小企業のユーザーが手軽に買えるように、その価格も低く設定されている。

ＰＣＳはバスアーキテクチャによって２つのファミリにわけることができる。第１のファミリはファミリ１モデルと呼ばれ、ＩＢＭ　ＰＣＡＴに代表されるようなＰＣＳを含む。第２のファミリ、すなわちファミリ２モデルは、ＩＢＭ　ＰＳ／２モデル５０ないし８０のように、マイクロチャネルバスアーキテクチャを使用する。

ファミリ１モデルの初期のＰＣＳでは、ソフトウエアの互換性が最も重要であると考えられていた。そのため、システム常駐コード（マイクロコード）という分離手段がハードウエアとソフトウエアの間に設定された。このコードは、ユーザーのアプリケーションプログラム／オペレーティングシステムと装置（デバイス）の間の動作インタフエースを与え、ハードウエアデバイスの特性についてユーザーが注意を払わなくてもよいようにしていた。最終的には、アプリケーションプログラムをハードウエアの特性から独立させつつ、新しいデバイスをシステムに付加できるようにするため、コードはＢＩＯＳへと発展していった。ＢＩＯＳは、特定のデバイスハードウエア特性に対するデバイスドライバの依存性をなくすと共に、デバイスへの中間インタフエースとしての働きをデバイスドライバに持たせたため、その重要性が直ちに認められた。ＢＩＯＳはシステムと一体化されていて、システムプロセッサに対するデータの入出力を制御するものであるから、読取専用メモリ（ＲＯＭ）の形でシステムプレーナに組み込まれていた。例えば、初期のＩＢＭパーソナルコンピュータのＢＩＯＳは、プレーナボード上のＲＯＭの８Ｋの領域を占めていた。

Ｃ．発明が解決しようとする課題

ＰＣＳの新しいモデルが開発される時は、新しいハードウエア及びＩ／Ｏ装置を導入するためにＢＩＯＳを拡張しなければならなかった。当然のように、ＢＩＯＳを記憶するためのメモリ容量は大きくなっていった。例えば、ＩＢＭパーソナルコンピュータＡＴの場合、ＢＩＯＳは３２Ｋバイトの ＲＯＭ容量を必要とした。

最近は新しい技術の発達により、ファミリー２モデルのＰＣＳは益々精巧になり、一般の消費者も使い始めている。技術の進歩は急速であり、新しいＩ／Ｏ装置がＰＣＳに使用されだしたので、ＰＣＳの開発サイクルでＢＩＯＳの変更が問題になってきた。

例えば、マイクロチャネルアーキテクチャを採用したＩＢＭパーソナルシステム／２の場合、新しい拡張ＢＩＯＳ（ＡＢＩＯＳ）が開発された。しかしソフトウエアの互換性を維持するため、フ

アミリー1モデルからのＢＩＯＳをファミリー2モデルに含ませておく必要があつた。ファミリー1のＢＩＯＳは、互換ＢＩＯＳ（ＣＢＩＯＳ）と呼ばれるようになつた。しかしＩＢＭパーソナルコンピュータＡＴに関連して説明したように、プレーナボード上には３２ＫバイトのＲＯＭしかなかつた。システムは９６ＫバイトのＲＯＭまで拡張できたが、システムの制約上、これがＢＩＯＳに対して使用できる最大の容量であつた。幸いなことに、ＡＢＩＯＳを追加しても、ＡＢＩＯＳ及びＣＢＩＯＳの両方を９６ＫのＲＯＭに詰め込むことができたが、残りの容量はごく僅かであつた。従つて、将来新しいＩ／Ｏ装置が付加されるときは、ＣＢＩＯＳ及びＡＢＩＯＳはＲＯＭスペースを超過してしまい、新しいＩ／Ｏ技術を容易にはＣＢＩＯＳ及びＡＢＩＯＳに組み込めなくなる。

上述のような問題に加えて、ファミリー2ＢＩＯＳの変更を開発サイクルの出来るだけ遅い段階で行いたいという要求があつたため、ＢＩＯＳの部分をＲＯＭから外すことが必要になつた。外された部分は、固定ディスクのような大容量記憶装置に記憶された。ディスクは読み書きが可能なため、実際のＢＩＯＳコードをディスク上で変更することが出来るようになつた。ディスクはＢＩＯＳコードを記憶する高速で効率の良い手段であるが、ＢＩＯＳコードが改悪される可能性が高くなつた。ＢＩＯＳはオペレーテイングシステムと一体化されているため、改悪されたＢＩＯＳは破壊的な結果を招き、多くの場合システムの完全な障害及びノーオペレーションを引き起こす。従つて、固定ディスクに記憶されているＢＩＯＳコードが許可なく変更されないようにするための手段が強く要求されており、そのような手段を提供することが本発明の目的の1つである。

Ｄ．課題を解決するための手段

本発明に従うＰＣＳは、システムプロセツサ、ランダムアクセスメモリ、ＲＯＭ、及び少なくとも1つの直接アクセス記憶装置を含んでいる。システムプロセツサと直接アクセス記憶装置の間に接続される直接アクセス記憶装置コントローラは、その記憶装置の一領域を保護するための手段を含む。保護される領域は、マスタブートレコード及びＢＩＯＳイメージを記憶する。保護手段はリセット信号に応答して保護領域へのアクセスを許可し、マスタブートレコードをランダムアクセスメモリへロードさせる。マスタブートレコードは動作時にＢＩＯＳイメージをランダムアクセスメモリにロードする。かくしてランダムアクセスメモリにあるＢＩＯＳが実行され、保護手段を活動化する信号を発生する。これにより、マスタブートレコード及びＢＩＯＳイメージを含むディスク上の領域へのアクセスが禁止される。次いでＢＩＯＳは、システムの動作を開始させるためオペレーテイングシステムをブートアップする。

ＲＯＭはＢＩＯＳの第1部分を含んでいる。この第1部分はシステムプロセツサ及び直接アクセス記憶装置を初期設定し、保護手段をリセットして、直接アクセス記憶装置の保護領域にあるマスタブートレコードをランダムアクセスメモリに読みこませる。マスタブートレコードはデータセグメント及び実行可能コードセグメントを含む。データセグメントはシステムハードウエアと、マスタブートレコードがサポートするシステム構成とを表わすデータを含む。第１ＢＩＯＳ部分は、マスタブートレコードのデータセグメントからのデータと、システムプロセツサ、システムプレーナ及びプレーナ１／Ｏ構成を表わす第１ＢＩＯＳ部分内のデータとが一致していることを検証することにより、マスタブートレコードがシステムハードウエアに合つていることを確かめる。

マスタブートレコードがシステムハードウエアに合つていれば、第１ＢＩＯＳ部分は、システムプロセツサにマスタブートレコードの実行可能コードセグメントを実行させる。実行可能コードセグメントは、システム構成が変わつていないことを確かめ、ＢＩＯＳの残りの部分を直接アクセス記憶装置からランダムアクセスメモリにロードする。次いで実行可能コードセグメントは、残余ＢＩＯＳ部分の真正さを検査し、システムプロセツサにランダムアクセスメモリにあるＢＩＯＳの実行を

開始させる。ランダムアクセスメモリ中のＢＩＯＳは、残りのＢＩＯＳを含むディスク区画を保護するための信号を発生し、オペレーティングシステムをブートアップして、ＰＣＳの動作を開始させる。残りのＢＩＯＳを含む区画を保護するのは、ディスク上のＢＩＯＳコードへのアクセスを禁止して、その完全性を守るためである。

Ｅ．実施例

図面、特に第２図において、複数のＩ／Ｏスロット１８を介してシステムボードないしプレーナボード２４に接続した複数のＤＡＳＤ（直接アクセス記憶装置）１２～１６を有するＰＣＳ（パーソナルコンピュータシステム）１０の一部を切除したものが示されている。電源２２は周知のようにシステム１０に電力を供給する。プレーナボード２４は、入力、プロセスおよび出力情報に対し計算機命令による制御のもとで動作するシステムプロセッサを含んでいる。

使用に当り、ＰＣＳ１０は主として少数のユーザーのグループまたはシングルユーザーに個々に計算パワーを与えるように設計されそして個人または小企業用に安価なものとされている。動作を述べると、このシステムプロセッサはＩＢＭのＯＳ／２またはＰＣ－ＤＯＳのようなオペレーティングシステムのもとで動作する。この形式のオペレーティングシステムはＤＡＳＤ１２～１６とオペレーティングシステムの間にＢＩＯＳインターフエースを含む。機能により複数のモジュールに分割されたＢＩＯＳの一部はプレーナ２４上のＲＯＭに記憶され、そして以降ではＲＯＭ－ＢＩＯＳと呼ぶ。ＢＩＯＳはハードウエアとオペレーティングシステムのソフトウエアの間にインターフエースを与え、プログラマまたはユーザが特定の装置について深いオペレーティング上の知識を必要とせずにそのマシンをプログラムしうるようにする。例えば、ＢＩＯＳディスケットモジュールはプログラマがディスケット駆動ハードウエアの深い知識がなくともディスケット駆動機構をプログラムしうるようにする。このように異る製造業者により設計され製造された多数のディスケット駆動機構がこのシステムで使用出来る。これはシステム１０のコストを低減するばかりでなくユーザが多数のディスケット駆動機構からそれを選ぶことが出来るようにする。

上記の構造を本発明に関連づける前に、ＰＣＳ１０の一般的な動作を要約する。第１図は、このＰＣＳ１０のブロック図である。第１図はプレーナ２４の構成要素と、Ｉ／Ｏスロット１８およびこのシステムの他のハードウエアへのプレーナ２４の接続を示している。プレーナ２４の上にはシステムプロセッサ２６が配置され、このプロセッサは局所母線２８により、メモリコントローラ３０に接続するマイクロプロセッサからなり、コントローラ３０はランダムアクセスメモリ（ＲＡＭ）３２に接続する。このマイクロプロセッサは適当なものでよいが、インテル社の８０３８６がその一例である。

本発明は以降において第１図について説明するが、本発明の装置および方法は他のプレーナボードハードウエア構成にも使用されるものである。例えば、システムプロセッサはインテル８０２８６または８０４８６でもよい。

このプロセッサによりプレーナ識別番号（プレーナＩＤ）がアクセス可能である。プレーナＩＤはそのプレーナに固有のものであり使用されているプレーナの型式を識別する。例えば、プレーナＩＤはシステムプロセッサ２６のＩ／Ｏポートを介して、またはスイッチを用いて読取るべくハードワイヤドしうる。また、ディスクコントローラへのプレーナ論理回路を用いて、システムプロセッサ２６の別のＩ／Ｏポートによりリセット信号を発生させることができる。例えば、リセット信号発生のためにそのＩ／Ｏポートをアドレスしてプレーナ論理を活動化するソフトウエアにより、リセット信号の発生を開始することができる。

局所母線２８は更に母船コントローラ３４を通じてプレーナ２４上の読取専用メモリ（ＲＯＭ）３６に接続する。

付加的な不揮発性メモリ（ＮＶＲＡＭ）５８が直列／並列ポートインターフエース４０および母

線コントローラ３４を介してマイクロプロセッサ２６に接続する。この不揮発性メモリ５８はシステムの電源が切れたときにも情報を維持するために電池でバックアップされたＣＭＯＳでよい。ＲＯＭは通常プレーナ上にあるから、ＲＯＭに記憶されたモデル値およびサブモデル値はシステムプロセッサおよびシステムプレーナＩ／Ｏ構成を夫々識別するために用いられる。このようにこれらの値はプロセッサとプレーナＩ／Ｏ構成を物理的に識別する。ＮＶＲＡＭ５８はシステム構成データを記憶するために用いられる。すなわち、このＮＶＲＡＭはシステムの現在の構成を記述する値を含む。例えばＮＶＲＡＭは固定ディスクまたはディスケットの容量、ディスプレイの型式、メモリ容量、時刻、日付等を記述する情報を含む。更に、ＲＯＭに記憶されたモデル値およびサブモデル値は構成設定のような特殊な構成プログラムが実行されるときにＮＶＲＡＭにコピーされる。構成設定プログラムの目的はシステムの構成を特徴づける値をＮＶＲＡＭに記憶させることである。このように適正に構成されたシステムについてのＮＶＲＡＭ内のモデル値およびサブモデル値はＲＯＭに記憶されたモデル値およびサブモデル値に夫々等しい。これらの値が等しくない場合には、それはそのシステムの構成が変更されていることを示す。第６Ｄ図により、ＢＩＯＳのローディングとノ組合せにおいてこの特徴を詳述する。

第１図の説明を続けると、母線コントローラ３４はＩ／Ｏプレーナ母線４３によりＩ／Ｏスロット１８、直列／並列インターフエース４０および周辺装置コントローラ４２に接続する。周辺装置コントローラ４２は更にキーボード４４、マウス４６、診断パネル４７、およびディスケットコントローラ６４に接続する。ＮＶＲＡＭ５８の他に、直列／並列インターフエース４０はプリンタ、ハードコピー装置等に情報を入／出力するための直列ポート４８と並列ポート５０に接続する。周知のように局所母線２８はキャッシュコントローラ５２、キャッシュメモリ６８、コプロセッサ５４およびＤＭＡコントローラ５６にも接続しうる。

システムプロセッサ２６はその内部動作並びにＰＣＳ１０の他のエレメントとのインターフエースを制御する。例えば図示のシステムプロセッサ２６は、固定ディスク駆動機構６２のようなＤＡＳＤに接続した小型コンピュータシステムインターフエース（ＳＣＳＩ）Ｉ／Ｏカード６０に接続する。ＳＣＳＩディスク駆動機構以外のものを本発明により固定ディスクとして使用出来る。固定ディスク６２に加えて、システムプロセッサ２６はディスケット駆動機構６６を制御するディスケットコントローラ６４にもインターフエースしうる。ここで「ハードファイル」は固定ディスク駆動機構６２であり、「フロッピー」はディスケット駆動機構６６を意味するものである。

本発明の以前にはＲＯＭ３６はハードウエア周辺装置に対しオペレーティングシステムをインターフエースするＢＩＯＳコードのすべてを含むことが出来た。本発明によればＲＯＭ３６はＢＩＯＳの一部のみを記憶するようにされる。この部分はシステムプロセッサ２６により実行されるとき、固定ディスク６２またはディスケット６６からＢＩＯＳの第２のすなわち残りの部分、以下ＢＩＯＳイメージと呼ぶ部分を入力する。このＢＩＯＳイメージは第１ＢＩＯＳ部分にとつて代わり、そしてこのシステムの一体化部分としてＲＡＭ３２のような主メモリ内に常駐する。ＲＯＭ３６に記憶されたＢＩＯＳの第１部分（ＲＯＭ－ＢＩＯＳ）は第３～４図で一般的に、そして第６Ａ～６Ｄ図で詳細に説明する。ＢＩＯＳの第２部分（ＢＩＯＳイメージ）は第５図で説明し、ＢＩＯＳイメージのローディングは第７図で説明する。ＤＡＳＤからのＢＩＯＳイメージのローディングによる他の利点はシステムプロセッサのＲＡＭ３２にＢＩＯＳを直接にロード出来るということである。ＲＯＭのアクセスはＲＡＭのそれより著しく高速であるから、コンピュータシステムの処理速度に大きな改善が得られる。

ＲＯＭ３６内のＢＩＯＳのオペレーションそして固定ディスクまたはディスケットからのＢＩＯ

Sイメージのローディングオペレーションを次に述べる。一般に、ROM－BIOSのような第1プログラムがシステムのプリチエツクを行い、そしてRAMにBIOSマスタブートレコードをロードする。マスタブートレコードは検証情報を有するデータセグメントと実行可能なコードを有しローディング手段として働くコードセグメントを含む。この実行可能なコードはハードウエアの適合性とシステム構成の妥当性を判断するためにデータ情報を使用する。ハードウエアの適合性と適正なシステム構成についてのテスト後に、この実行可能なコードがRAMにBIOSイメージをロードし、主メモリ常駐プログラムを生成する。BIOSイメージはROM－BIOSに続き、そしてマシンのオペレーションを開始するためにオペレーティングシステムをロードする。説明の便宜上、マスタブートレコードの実行可能コードセグメントをMBRコード、データセグメントをMBRデータと呼ぶ。

第3図はROM－BIOSを構成している異つたコードモジユールを示すメモリマツプを示す。ROM－BIOSは電源投入自己検査（POST）ステージIモジユール70、初期BIOSロード（IBL）ルーチンモジユール72、デイスケツトモジユール74、ハードフアイルモジユール76、ビデオモジユール78、診断パネルモジユール80およびハードウエア適合性データ82を含む。要するに、POSTステージI70はシステムの事前初期設定とテストを行う。IBLルーチン72はBIOSイメージがデイスクまたはデイスケツトからロードされるべきかを決定し、適合性をチエツクし、そしてマスタブートレコードをロードする。デイスケツトモジユール74はデイスケツト駆動機構についての入出力機能を与える。ハードフアイルモジユール76は固定デイスク等へのI／Oを制御する。ビデオモジユール78はビデオデイスプレイに接続するビデオI／Oコントローラへの出力機能を制御する。診断パネルモジユール80はシステム用の診断デイスプレイ装置に対する制御を与える。ハードウエア適合性データ82は第5図で述べるシステムモデル値およびサブモデル値のような値を含む。

第4図は固定デイスクまたはデイスケツトからシステムにBIOSイメージをロードするためのプロセス全体を示す。システムに電源が投入されると、システムプロセツサがステツプ100においてPOSTステージIの入口点へと導かれる。POSTステージIはシステムを初期設定し、選択されたDASDからのBIOSイメージのロードに必要なシステム機能のみをステツプ102でテストする。特にPOSTステージIは、必要であればプロセツサ／プレーナ機能、診断パネル、メモリサブシステム、割込みコントローラ、タイマー、DMAサブシステム、固定デイスクBIOSルーチン（ハードフアイルモジユール76）およびデイスケツトBIOSルーチン（デイスケツトモジユール74）を初期設定する。

POSTステージIがシステムを事前初期設定した後に、POSTステージIは初期BIOSロード（IBL）ルーチンにシステムプロセツサを導く。このIBLルーチンはまずBIOSイメージが固定デイスクに記憶されているかあるいはデイスケツトからロードされるかを決定し、次にステツプ104において選択された媒体（デイスクかデイスケツトか）からRAMにマスタブートレコードをロードする。マスタブートレコードはMBRデータとMBRコードを含む。MBRデータは検査用であり、MBRコードはBIOSイメージのロードのために実行される。IBLルーチンの詳細は第6A～6D図に示してある。

第4図において、IBLルーチンがマスタブートレコード（MBR）をRAMにロードした後に、ステツプ106において、システムプロセツサがMBRコードのスタートアドレスを用いて実行を開始する。MBRコードはBIOSイメージの確認とシステム構成の検査のための一連の妥当性テストを行う。このMBRコードの詳細は第7図に示してある。

これら妥当性テストにもとづき、MBRコード

はＢＩＯＳイメージをＲＡＭにロードし、そしてステップ１０８において制御を主メモリに新しくロードされたＢＩＯＳイメージへと移す。特に、ＢＩＯＳイメージは前にＲＯＭ－ＢＩＯＳが占めていたアドレススペースにロードされる。すなわちＲＯＭ－ＢＩＯＳが０００００ＨからＦＦＦＦＦＨまでの間でアドレスされるとすると、ＢＩＯＳイメージはこのＲＡＭアドレススペースにロードされ、かくしてＲＯＭ－ＢＩＯＳに代えられる。次に制御は新しくロードされたＢＩＯＳイメージに含まれるＰＯＳＴステージＩＩに移され、かくしてＲＯＭ－ＢＩＯＳは放棄される。この後、ＲＡＭにあるＰＯＳＴステージＩＩはオペレーティングシステムをロードするためのステップ１１０～１１４を実行する。オペレーティングシステムに制御を移す前にステージＩＩＰＯＳＴは、ＢＩＯＳイメージを保持するディスク区画へのアクセスを禁止するための保護手段をセットする。保護の詳細は第８～１０図のところで説明する。ウオームスタート中にプロセッサはステップ１００～１０６をバイパスしてステップ１０８に導かれる。

ここでマスタブートレコードの形式について説明しておく。第５図はマスタブートレコードの形式を示す。ブートレコードは実行可能コードセグメント１２０とデータセグメント１２２～１３８を含む。ＭＢＲコード１２０はＲＯＭ－ＢＩＯＳの識別検査、ＩＢＬブートレコードがシステムと両立しうるかどうかのチエツク、システム構成の検査および選ばれたＤＡＳＤ（ディスクまたはディスケット）からのＢＩＯＳイメージのロードを実行するためのＤＡＳＤに依存するコードを含む。データセグメント１２２～１３８は媒体の定義、マスタブートレコードの識別と検査、ＢＩＯＳイメージの位置決めおよびＢＩＯＳイメージのロードに用いられる情報を含む。

マスタブートレコードはブートレコード識別子１２２により識別される。ブートレコード識別子１２２はレコードのはじめの３バイト内の文字列「ＡＢＣ」のような固有のビットパターンである。マスタブートレコードの保全性は、ブートレコードがロードされるとき計算された検査合計値と比較される検査合計値１３２によりテストされる。データセグメントは更に少くとも１個の適合プレーナＩＤ値１３４、適合モデル値およびサブモデル値１３６を含む。マスタブートレコードのプレーナＩＤ値はマスタブートレコードが有効なプレーナを規定する。同様にマスタブートレコードのモデル値およびサブモデル値はマスタブートレコードが有効であるプロセッサとプレーナＩ／Ｏ構成を夫々規定する。ブートレコード識別子と検査合計は有効なマスタブートレコードを識別し、そしてブートレコードのプレーナＩＤ、モデル値およびサブモデル値の比較は、システムに適合するブートレコードの識別とシステム構成の有効性の決定に用いられる。他の値であるブートレコードパターン１２４はＲＯＭ－ＢＩＯＳの妥当性の決定に用いられる。ブートレコードパターン１２４はＲＯＭに記憶された対応するパターン値と比較される。それらが一致することは、有効なＲＯＭ－ＢＩＯＳが選ばれた媒体からのＢＩＯＳイメージのロードを開始したことを示す。

以下にマスタブートレコードの夫々の値とそれらの機能を詳述する。

ＭＢＲ識別子（１２２）：　ＩＢＬブートレコードのはじめの３バイトは「ＡＢＣ」のような文字からなりうる。これはブートレコードの識別に用いられる。

ＭＢＲコードセグメント（１２０）：　このコードは対応するプレーナＩＤとモデル／サブモデル値の比較により、プレーナおよびプロセッサに対するブートレコードの適合性を検査する。これらの値の一致は選ばれた媒体からシステムＲＡＭへのＢＩＯＳイメージのロードを生じさせる。システムイメージ（メモリにロードされたＢＩＯＳイメージ）検査合計が有効であり媒体ロードエラーが生じないならば、ＭＢＲコードは制御をシステムイメージのＰＯＳＴステージＩＩルーチンに移す。

ＭＢＲパターン（１２４）：　ＩＢＬブートレコードデータセグメントのはじめのフイールドは文字列「ＲＯＭ－ＢＩＯＳ　１９８９」のようなパター

ンを含む。この文字列は、ブートパターン値とＲＯＭ内に記憶された対応する値（ＲＯＭ－パターン）を比較することによりＲＯＭ－ＢＩＯＳの妥当性チエツクに用いられる。

ＭＢＲ版デート（１２６）：　マスタブートレコードは更新のための版デートを含む。

システム区画ポインタ（１２８）：　データセグメントはステージＩＩＰＯＳＴで用いるための媒体システム区画領域の開始点を示す媒体ポインタを含む。ＩＢＬディスケツトではこのポインタはトラツクーヘツドーセクタ形式であり、ディスクでは相対ブロツクアドレス（ＲＢＡ）形式である。

システム区画タイプ（１３０）：　システム区画タイプは媒体システム区画の構造を示す。３つのシステム区画構造のタイプ、すなわち、完全、最小および不在、がある。完全システム区画はＢＩＯＳイメージおよびマスタブートレコードに加えてセツトアツプユーテイリテイおよび診断を含む。最小システム区画はＢＩＯＳイメージとマスターブートレコードのみを含む。システムがＩＢＬイメージを有するハードファイルへのアクセスを有さないことがあり、この場合にはシステム区画タイプは不在となる。この例ではＩＢＬはディスケツトから生じることになる。これら三種のシステム区画タイプにより、媒体上でシステム区画が占めるスペースを変えることができる。

検査合計値（１３２）：　データセグメントの検査合計値は、ＭＢＲコードのレコード長値（１．５ｋバイト）についての有効検査合計を発生するために初期設定される。

ＭＢＲプレーナＩＤ値（１３４）：　データセグメントは適合プレーナＩＤを規定するワード列のような値を含む。各ワードは１６ビツトのプレーナＩＤからなり、この列はワード値０で終る。システムのプレーナＩＤがマスタブートレコード内のプレーナＩＤ値と一致すると、ＩＢＬ媒体イメージはシステムプレーナに適合しうる。システムのプレーナＩＤが列内のいずれのワードとも一致しないならば、ＩＢＬ媒体イメージはシステムプレーナに適合しない。

ＭＢＲモデル値およびサブモデル値（１３６）：　このデータセグメントは適合プロセツサを規定するワード列のような値を含む。各ワードはモデル値およびサブモデル値からなり、このワード列はワード値０で終了する。システムのモデル値およびサブモデル値（ＲＯＭに記憶されたもの）がこのワード列内の１つのワードと一致するならばＩＢＬ媒体イメージはそのシステムプロセツサに適合する。ＲＯＭモデル値とＲＯＭサブモデル値がこのワード列中のいずれのワードとも一致しないならば、ＩＢＬ媒体イメージはシステムプロセツサに適合しない。

ＭＢＲマツプ長（１３８）：　ＩＢＬマツプ長は媒体イメージブロックの数に初期設定される。云い換えると、ＢＩＯＳイメージが４個のブロツクに分割されるならばマツプ長は４となり、４個のブロツクポインタ／長さフイールドを示す。媒体イメージは１個の連続した１２８ｋブロツクであるから一般にこの長さは１にセツトされる。

ＭＢＲ媒体セクタサイズ（１３８）：このワード値は媒体セクタサイズ（セクタのバイト数）に初期設定される。

媒体イメージブロツクポインタ（１３８）：　この媒体イメージブロックポインタは媒体上でシステムイメージブロツクを見つけ出す。通常は、媒体イメージが１個の連続したブロツクとして記憶されるため、１個のポインタしか存在しない。ＩＢＬディスケツトではポインタはトラツクーヘツドーセクタ形式となつており、ディスクでは相対ブロツクアドレス形式となつている。

媒体イメージブロツク長（１３８）：　媒体イメージブロツク長は対応するイメージブロツクポインタが示すブロツクのサイズ（セクタ数）を示す。ＢＡＳＩＣ用のスペースを含む１２８ｋの連続する媒体イメージの場合には、このフイールドは２５６にセツトされて、ＢＩＯＳイメージブロツクが媒体イメージブロツクポインタ位置からはじまり２５６個のセクタ（５１２バイト／セクタ）からなることを示す。

第６Ａ～６Ｄ図はＩＢＬルーチンの詳細なフロー

チャートである。通常状態ではＩＢＬルーチンはシステムの固定ディスクからＲＡＭの特定のアドレスにマスタブートレコードをロードし、そしてシステムプロセッサをこのマスタブートレコードのコードセグメントの実行開始へと導く。またＩＢＬルーチンは、マスタブートレコードがディスケットからロードされうるディスケットデフォルトモードを含む。しかしながら、ＩＢＬルーチンはシステムが固定ディスク上にＩＢＬ媒体を含みそして有効パスワードがＮＶＲＡＭにあるならば、ディスケットデフォルトモードの実行を許可しない。ユーザーはＮＶＲＡＭにそのパスワードをセットしうる。ディスケットデフォルトモードを禁止する目的は、ディスケットからの許可されないＢＩＯＳイメージのロードを防止することである。云い換えると、ディスケットデフォルトモードはシステムの固定ディスクが動作不能であり、ユーザがディスケットからのロードを望むことを示す（パスワードをセットしない）ときにのみ用いられる。ＩＢＬルーチンがいずれの媒体からのマスタブートレコードのローディングもしえないならば、エラーメッセージが発生されてシステムが停止する。

第６Ａ図において、通常、システムはシステム固定ディスクを含み、このディスクがＩＢＬルーチンにより初期設定される（ステップ１５０）。固定ディスクがＰＣＳのドライブＣ用の構成とされているものとし、またドライブＡがディスケット駆動機構に割りふられているとすると、ＩＢＬルーチンはドライブＣがＩＢＬ媒体を含むかどうかの決定をステップ１５２で行う。このプロセスの詳細を第６Ｂ図に示す。ＩＢＬルーチンは固定ディスクの読取りを最後の３セクタのところから開始し、９９セクタの間、あるいは有効マスタブートレコードが見い出されるまで媒体ポインタを減分しながら読取りを続ける。マスタブートレコードが見い出されたならば、ステップ１５６でシステムプレーナおよびプロセッサに対する適合性についてチエックする。適合性がなければステップ１５８でエラーが出される。ステップ１５２において固定ディスク（１次ハードファイル）の最後の９９セクタにマスタブートレコードがなければ、ステップ１５４でエラーが出される。

ステップ１５６でマスタブートレコードがあれば一連の有効性チエックが行われ、マスタブートレコードがコンピュータシステムに適合しているかどうかを決定する。更に、このシステムの構成がチエックされる。このプロセスの詳細を第６Ｄ図に示す。ブートレコードがプレーナＩＤ、モデルおよびサブモデルに適合し、そして更にシステム構成が変更されていなければ、ステップ１６０でマスタブートレコードがロードされ、そしてそのコードセグメントが実行される。

ステップ１５４と１５８において、固定ディスクからマスタブートレコードをロードする際にエラーが生じあるいは固定ディスクが使用出来ない場合には、ＩＢＬルーチンはステップ１６２で有効パスワードがＮＶＲＡＭに含まれているかどうかを決定する。このパスワードはＢＩＯＳイメージがディスケットからロードされうるかどうかを決定する。このパスワードは、セットアップユーティリティを動かすユーザにより設定されているときにのみ存在する。パスワードがＮＶＲＡＭに設定されていれば、ステップ１６４においてＢＩＯＳイメージはディスケットからロードされないようにされる。これはユーザに、固定ディスク上のＢＩＯＳイメージだけをシステムにロードしうるようにすることによりシステムのオペレーションの完全性を保証しうるようにする。このパスワードはＮＶＲＡＭに記憶された文字列の形をとることが出来る。

ステップ１６２において、ＮＶＲＡＭ内に有効パスワードがなく、ＢＩＯＳイメージがディスケットからロードしうる場合には、ＩＢＬルーチンがステップ１６６においてディスケットサブシステムを初期設定する。このＩＢＬルーチンはステップ１６８においてドライブＡがディスケット上にＩＢＬ媒体を含むかどうかを決定するドライブＡがＩＢＬ媒体を含んでいなければステップ１７０において無効ディスケットがドライブに挿入さ

れていることをユーザに知らせるためのエラーが発生する。ステップ１６８の詳細を第６Ｃ図に示す。

ステップ１６８において、ドライブＡがＩＢＬ媒体についてチエツクされた後に、ステップ１６０でマスタブートレコードがＲＡＭにロードされ、そしてそのコードセグメントが実行される。ディスケットについては、ＩＢＬルーチンは固定ディスクシステムに用いた妥当性チエツクを含まない。例えばシステムに新しいプロセツサが加えられる場合には、新しいＢＩＯＳイメージがディスケットに含まれることになる。新しいプロセツサは固定ディスクからのローデイングについて妥当性エラーを生じさせるから、ＩＢＬルーチンはＢＩＯＳイメージをディスケットからロードすることによりこれらテストをバイパスする能力を与える。

要点を繰返すと、マスタブートレコードはブートレコード値に対するシステムプレーナＩＤとプロセツサモデル／サブモデル値の整合により、システムとの適合性についてチエツクされる。ディスクについてはこのチエツクはまずＩＢＬルーチン７２で行われ、次にＩＢＬブートレコードにおいて再び行われる。最初のチエツク（ＩＢＬルーチンにおける）はブートレコードがシステムに適合することを確認するために行われ、２番目のチエツク（ブートレコードにおける）は適合ＲＯＭが制御をブートレコードに移したことを確認するために行われる。ディスケットブートレコードで行われるチエツクは、ＩＢＬルーチンが適合性をすでにチエツクしているから、適合ＲＯＭについては誤りはない。一方、最初の適合性チエツクはディスケットについては行われない。プレーナ／プロセツサの適合性はディスケットブートレコードの実行中にのみチエツクされる。この方法により、基準ディスケットからの新しいＢＩＯＳイメージのローデイングにおける将来の変更が可能になる。

第６Ａ図のＩＢＬルーチンにおける妥当性テストを更に詳細に説明する。第６Ｂ図は第６Ａ図における有効マスタブートレコードがドライブＣにあるかどうかの決定のためのステップ１５２の詳細なフローチヤートである。このプロセスはステップ２００でＩＢＬルーチンがドライブＣにアクセスしうるようにするドライブパラメータを得ることにより開始する。ＩＢＬロード位置がステップ２０２においてデイスクから最後の３セクタ（これらセクタは通常マスタブートレコードを含む）にセットされる。デイスクからマスタブートレコードを読取る試行回数を示すロードカウントがステップ２０４において１にセツトされる。ＩＢＬロード位置の３個のセクタがステップ２０６においてデイスクから読取られる。ステップ２０８～２１０において、何らかのデイスクドライブエラーが検出され、デイスク読取エラーが生じればそれが報告される。プロセスはエラー標識を伴つてリターンすることになる（ステップ２１２～２１４）。

ステップ２０８でドライブエラーが生じていなければ、そのデイスクレコードがステップ２１６においてマスタブートレコード識別子について走査される。文字「ＡＢＣ」のようなこのブートレコード識別子はデイスクレコードのはじめの３バイトと比較される。デイスクレコードが有効ブートレコード識別子（文字「ＡＢＣ」）を有し、そしてメモリにロードされたデイスクレコードから計算された検査合計がブートレコードの検査合計に等しければ、このデイスクレコードはステップ２１８においてエラーを有さない有効ブートレコードとして示される。ステップ２１４においてこのプロセスは第６Ａ図に戻る。

ステップ２１６においてブートレコード識別子または検査合計が無効であれば、ロードカウントがステップ２２０において１だけ増加される。このロードカウントはステップ２２２において９９のような予定の定数と比較される。ブートレコードの読取を９９回試み、不成功である場合にはステップ２２４、２１２および２１４においてエラーが示されリターンする。ブートレコードの読取りが９９回より少いときはＩＢＬロード位置がステップ２２６で１だけ減分され、そして３個の新し

いセクタがステップ２０６で新しいロード位置から読取られる。かくして最後の９９個のセクタ（３３コピーに等価）から有効ＩＢＬブートレコードがロードされえないときにはエラー条件がセットされて、制御がＩＢＬルーチンに戻される。

マスタブートレコードをドライブＡのディスケットからロードすることについての詳細を示す第６Ｃ図において、まずドライブＡのアクセスのための、ディスケットドライブパラメーターがステップ２３０でとり出される。ＩＢＬロード位置がステップ２３２においてディスケットの最後の３個のセクタ（シリング、ヘッドおよびセクタ形式）セットされる。これら３個のセクタがステップ２３４で読取られる。ステップ２３６～２３８において、ディスケットドライブエラーが検出されるとエラーが示される。ステップ２４０～２４２において、エラー条件がセットされ、制御がＩＢＬルーチンに戻される。

ステップ２３６においてドライブエラーが検出されないと、ディスケットレコードはステップ２４４においてブートレコード識別子をチエックされ、そして検査合計が計算される。ブートレコード識別子がなくあるいは検査合計が無効であれば、エラーが示されて制御がＩＢＬルーチンにもどされる。有効ブートレコード識別子と有効検査合計が検出されると標識がセットされ（ステップ２４８）、そして制御がＩＢＬルーチンに戻される。ディスケットロードの場合、ＩＢＬルーチンは固定ディスクロードにおけるようなセクタの探索を行わない。それ故ディスケットロードでは、ＩＢＬ媒体はディスケットの特定の位置に記憶されねばならない。

最後に、第６Ｄ図はシステムプレーナとプロセッサの適合性および適正なシステム構成についてのＩＢＬルーチンにおけるテスト方法を示す。ステップ２６０においてマスタブートレコードが、そのプレーナＩＤ値をシステムプロセッサにより読取られたシステムプレーナＩＤと比較することにより、システムプレーナとの適合性をチエックされる。システムプレーナＩＤがブートレコードプレーナＩＤ値と一致しないときは、このマスタブートレコードがこのプレーナに適合しないことを示す。ステップ２６２、２６４、２６６において、エラーが示され制御はＩＢＬルーチンに戻る。

マスタブートレコードがプレーナと適合するのであれば、ステップ２６８においてプロセッサとの適合性についてマスタブートレコードがチエックされる。ブートレコードのモデル値およびサブモデル値がＲＯＭに記憶されたモデル値およびサブモデル値と夫々比較される。一致しないときは、新しいプロセッサが多分導入されており、このブートレコードがその新しいプロセッサに適合しないことを示す。ステップ２７０、２６４、２６６において、エラーが示されそして制御はＩＢＬルーチンに戻る。マスタブートレコードがプレーナおよびプロセッサに適合するのであれば、ステップ２７２でＮＶＲＡＭが信頼出来るかどうかについてのチエックを行う。ＮＶＲＡＭが信頼出来なければ、ステップ２７４、２６６において、エラーが示されそして制御がＩＢＬルーチンに戻る。ＮＶＲＡＭが信頼出来れば、ステップ２７６においてシステム構成がチエックされる。ＮＶＲＡＭに記憶されたモデル値およびサブモデル値がＲＯＭに記憶されたモデル値およびサブモデル値と一致しない場合は、システム構成が変更したことを示す。この最後の比較は構成エラーのみを示す。構成エラーが生じると、ユーザに対しエラーが知らされる。このエラーは、構成設定を最後にランさせた後にシステム構成が変更されたことをユーザに示すものである。ステップ２７８、２６４、２６６において、ユーザは変更された構成についての情報を受け、そして制御がＩＢＬルーチンに戻される。このエラーは致命的なものではなく、構成設定（構成プログラム）を実行すべきことをユーザに知らせるものである。ステップ２７６でシステムモデル／サブモデル値が一致すると、ステップ２７４で適合性標識がセットされ、ＩＢＬルーチンに戻る。このように、マスタブートレコードとシステムの適合性は、システム構成が変化したかどうかの決定と並行してテストされる。

ＩＢＬルーチンがマスタブートレコードをＲＡＭにロードした後に、制御がＭＢＲコード開始アドレスに移される。第７図において、マスタブートレコードの実行可能コードセグメントがまずステップ３００においてＲＯＭおよびブートレコードのパターンを検査する。マスタブートレコード内のパターンがＲＯＭ内のパターンと一致しない場合にはステップ３０２においてエラーが発生され、ステップ３０５でシステムは停止する。ＲＯＭとブートレコードのパターンの一致についてのチエツクは、ディスクまたはディスケツトからロードされたマスタブートレコードがプレーナボード上のＲＯＭに適合しているかどうかを確認する。ステップ３００でＲＯＭのパターンがブートレコードのそれと一致すれば、ＭＢＲコードがステップ３０４でシステムプレーナＩＤ値、モデル値およびサブモデル値を対応するマスタブートレコード値と比較する。このプロセスは第６Ｄ図において述べた。これら値が一致しない場合には、マスタブートレコードがシステムプレーナおよびプロセッサと適合しないこと、あるいはシステム構成が変わっていることを示し、ステップ３０６でエラーが発生される。そのときステップ３０５でシステムは停止する。

ステップ３０４においてシステムプレーナＩＤ値、モデル値およびサブモデル値が対応するマスタブートレコード値と一致すると、ステップ３０８でＭＢＲコードが選ばれた媒体からシステムＲＡＭにＢＩＯＳイメージをロードする。ステップ３１０でデータ読取において媒体ロードエラーが生じると、ステップ３１２においてエラーが発生されてシステムが停止する。ステップ３１０で媒体ロードエラーが発生しないと、ステップ３１４で検査合計がメモリ内のＢＩＯＳイメージについて計算される。この合計が無効であれば、ステップ３１８でエラーが発生されてシステムが停止する。ステップ３１６での検査合計が有効であれば、ステップ３２０でシステム区画ポインタが記憶され、そしてステップ３２２でシステムプロセッサがＰＯＳＴステージＩＩで移されてシステムのローディングを開始する。

第８図は、ディスク駆動機構３５１とシステムプロセッサの間のデータ転送を制御する知能ディスクコントローラ３５０を示している。ディスクコントローラ３５０は第１図のアダプタカード６０に組み込むことができるが、その場合は、ディスク駆動機構３５１は固定ディスク駆動機構６２に対応することになる。本実施例のディスクコントローラ３５０は、本出願人が製造しているＳＣＳＩアダプタ（部品番号３３Ｆ８７４０）である。ディスクコントローラ３５０に含まれるマイクロプロセッサ３５２は自身の内部クロツクに従つて動作し、その内部オペレーションだけでなく、ディスクサブシステムの他の構成要素およびシステムプロセッサとのインターフエースも制御する。マイクロプロセッサ３５２は命令母線３５４によりＲＯＭ３５６に接続される。ＲＯＭ３５６は、ディスク駆動機構とシステムプロセッサの間のデータ転送の処理および制御のためにディスクコントローラ３５０が実行する命令を記憶している。マイクロプロセッサ３５２には、データの記憶や検索のためのランダムアクセスメモリを接続することが出来る。ディスクコントローラ３５０とシステムプロセッサの間のデータ転送はデータ母線３５８および命令母線３６０により行われる。線３６２上のリセット信号は、電源投入時あるいはシステムリセット時にディスクコントローラ論理をリセットないし初期設定する。このリセット信号はプレーナボード論理で発生され、例えば本出願人が１９８７年５月に発行した'ＩＢＭ ＰＥＲＳＯＮＡＬ ＳＹＳＴＥＭ／２ Ｓｅｍｉｎａｒ Ｐｒｏｃｅｅｄｉｎｇｓ"第５巻、第３号に記載されているようなマイクロチヤネルアーキテクチヤのチヤネルリセット信号の形をとりうる。また、プレーナ論理が接続されているシステムプロセッサのＩ／Ｏポートにビットが特定のビット構成を出力することによってリセット信号を効果的に開始することが出来る。

周知のように、マイクロプロセッサ３５２は、ディスク駆動機構とシステムプロセッサの間で効

率の良いデータ転送を行うためのすべてのインターフエース信号及びタイミング信号を供給する。ここでは簡単のため、本発明の理解に必要な信号だけを取り上げる。また、ＲＯＭ３５６に記憶されているプログラムについても、本発明の理解に必要なものだけを第９図を参照しながら説明する。

第９図はディスクコントローラの読取り、書込みおよび保護の機能を示すフローチヤートで、ＲＯＭ３５６に記憶されているルーチンにより制御される。動作時には、ディスク命令がシステムプロセツサからディスクコントローラへ送られる。ディスクコントローラはステツプ４００でこの命令を受け取つて解釈し、指定されたオペレーションを実行する。ディスクコントローラは、まずステツプ４０２で、オペレーシヨンがシステムプロセツサからディスクへのデータ書込みかどうかを調べる。もし書込みであれば、相対ブロックアドレス（ＲＢＡ）形式でシステムプロセツサからデータを受け取る。

説明を続ける前にＲＢＡ形式について簡単に触れておく。ＲＢＡは、ディスクのような大容量記憶装置のデータが所定サイズのブロックにおいてシーケンシヤル番号でアドレス指定されるような方式のことである。例えばブロックサイズを１０２４バイトとすると、ディスク容量が１０メガバイトの場合、システムプロセツサは約１００００のブロックをアドレス指定出来る。本発明に従うＰＣＳ用のオペレーテイングシステムにおいてＲＢＡを用いると、大容量記憶装置を極めて高速に且つ効率良くアドレス指定することが出来る。

説明の便宜上、次のことを仮定する。第１に、ディスクは全部でＮ個のブロックをサポート出来る。第２に、システムプロセツサはブロックＫを転送する。Ｋは０以上、Ｎ－１以下である。第３に、ディスクコントローラは最大アドレス可能ブロックＭを設定することが出来る。これは、ＫがＭより小さい場合にデータブロックへのアクセスを許可し、ＫがＭ以上の場合にデータブロックへのアクセスを拒否する。従つて、ＭをＮよりも小さい値に設定すると、ＭからＮ－１までのブロックについてディスク上に保護領域が生成される。以下に述べるように、この特徴によりＩＢＬ媒体が保護される。

第９図に戻つて、ステツプ４０４でデータがＲＢＡ形式で受け取られる。次にディスクコントローラは、ステツプ４０４において、受け取つたブロックの番号Ｋが最大ブロック値Ｍ（＜Ｎ）より小さいかどうかを調べる。ＫがＭより小さければ、ステツプ４０８でディスクコントローラはＲＢＡ形式をシリンダーヘツドーセクタ（ＣＨＳ）形式のような大容量記憶装置特有の形式に変換する。例えば、ディスクコントローラは表引きを利用してＲＢＡアドレスを一意的なＣＨＳ記憶位置に変換することが出来る。変換式を利用してＲＢＡをＣＨＳに変換する方法もある。例えば、１つのヘツド、６４のシリンダおよび９６のセクタを有するディスクの場合、ヘツド＝０、シリンダ＝ＲＢＡ／９６の商、セクタ＝ＲＢＡ／９６の剰余となる。ＲＢＡ形式をＣＨＳ形式に変換した後、ステツプ４１０でデータがディスクの変換されたＣＨＳ記憶位置に書込まれる。次にディスクコントローラはステツプ４１２に進んで、システムプロセツサからの別の命令を待つ。

ステツプ４０６に戻つて、受け取つたＲＢＡが設定された最大ＲＢＡ値以上であれば、ステツプ４１４でアクセスが拒否される。すなわち、ＫがＭ以上であれば、ブロックＫはディスクには書込まれない。したがつて、もしＩＢＬ媒体がＭからＮ－１までのブロックに記憶されていると、そのＩＢＬ媒体は書込みから保護されることになる。

ステツプ４０２に戻つて、システムプロセツサからの命令が書込みでなければステツプ４１６で読取命令かどうかを調べる。読取命令であれば、システムプロセツサは要求されたデータについてのＲＢＡ形式を送つているから、ステツプ４１８でそれを受け取る。後続のステツプ４２０、４２２、４２４および４２６は、読取が行われることを除くと、前述のステツプ４０６、４０８、４１０および４１４と同じである。従つて、ＩＢＬ媒体はコピーからも保護される。読取られたデータ

はステップ４１２でシステムプロセッサに転送される。

ステップ４１６に戻って、命令が読取でなければ、ステップ４２８で最大ＲＢＡ設定命令かどうかを調べる。ディスクコントローラは、この命令により、ディスク上に保護領域ないし保護区画を生成することが出来る。この命令を受け取ると、ディスクコントローラはステップ４３０でＭを０からＮまでの間の値に設定する。ディスクコントローラが（リセット信号により）リセットされると、Ｍは最大数のブロックが使用可能になるように設定される。すなわち、ディスクコントローラがリセットされたときは、Ｍ＝Ｎである。基本的には、特定領域の保護は、ディスクコントローラがリセットされると解除され、その領域へのアクセスが可能になる。最大ＲＢＡ設定命令が実行されてしまうと、リセットまたは別の最大ＲＢＡ設定命令に依らない限り、保護領域へのアクセスは行えない。次いでディスクコントローラはステップ４１２へ進んで、別の命令を待つ。

ステップ４２８に戻って、命令が最大ＲＢＡ設定でなければ、ステップ４００で受け取った命令は、書込み、読取および最大ＲＢＡ設定以外の命令であり、ステップ４３２で実行される。そのような命令は本発明とは無関係であるから、ここでは触れないことにする。ディスクコントローラはステップ４１２へ進み、別の命令を待つ。

以上のことを考慮しながら、次にＩＢＬ媒体のローディングおよび保護のオペレーションについて説明する。一般に、ＩＢＬ媒体を有するディスクコントローラは、コールドスタート（電源投入）またはウオームスタート（Ａｌｔ－Ｃｔｒｌ－Ｄｅｌ）でリセットされる。これにより、最大ＲＢＡ（Ｍ）はＮに設定され、ＩＢＬ媒体へのアクセスが可能になる。こうしておかないと、システムはオペレーションを開始するためにＩＢＬ媒体をロードすることが出来なくなる。一旦ＩＢＬ媒体がロードされて実行されると、ディスク上のＩＢＬ媒体へのアクセスを禁止するために、最大ＲＢＡがＩＢＬ媒体より下に設定される。

第１０図はＩＢＬ媒体の保護に関するフローチャートを示している。システムはステップ４５０の電源投入により初期設定され、ＢＩＯＳがプレーナボード論理で活動を開始して、ステップ４５２でディスクコントローラにリセット信号を送る。このリセット信号はＩＢＬ媒体の保護を解除し、システムプロセッサがそれをアクセス出来るようにする。ＩＢＬ媒体はブロックＭからブロックＮ－１までの領域で記憶されている。システムは第４～７図のところで説明したようにして、ステップ４５４でＩＢＬ媒体をロードする。ＩＢＬローディングシーケンスの間にステップ４５６でＰＯＳＴステージIIが実行される。ＰＯＳＴステージIIのタスクの１つは、最大ＲＢＡ設定命令を実行して、最大ＲＢＡをＩＢＬ媒体の最初のブロックの番号Ｍに設定することである（ステップ４５８）。Ｍは前に説明した区画のタイプ（完全、最小、不在）に依存する。これによりＩＢＬ媒体へのアクセスは禁止されるが、ディスクの他の領域へのアクセスは可能である。最後のステップ４６０でオペレーティングシステムがブートアップされる。

ステップ４６２のウオームスタートの場合、プレーナ論理は次のステップ４６４で、ＰＯＳＴステージIIによりディスクコントローラをリセットするよう指令される。これはＩＢＬ媒体の保護を解除する。この場合は、ＩＢＬ媒体は既にＲＡＭにあるので、それが再びロードされることはない。しかしＩＢＬ媒体の保護が解除されているため、ＰＯＳＴステージIIを実行することによって再び保護を図る必要がある。そのためステップ４５８が実行され、ステップ４６０でシステムが再ブートされる。

要約すると、本発明は固定ディスクのような大容量記憶装置に記憶されているＩＢＬ媒体へのアクセスを保護するための方法および装置を提供するものである。ＩＢＬ媒体の保護は、大容量記憶装置をブロック単位でアドレス指定し、システムが通常のオペレーション中にアクセス出来る最大ブロック番号を設定することで実現される。ＩＢＬ媒体は、それよりも番号の大きいブロックのと

ころに連続的に記憶される。ディスクコントローラに送られるリセット信号は、最大ブロック値を解除し、システムがＩＢＬ媒体をアドレス指定できるようにする。リセット信号は、電源投入時またはウォームスタート時に発生される。

Ｆ．発明の効果

本発明によれば、ディスク等の直接アクセス記憶装置に記憶されているＢＩＯＳが許可なく変更されるのを効果的に防止することができる。

4．図面の簡単な説明

第１図は、本発明に従うＰＣＳの構成を示すブロック図。

第２図は、ＰＣＳ本体の破断斜視図。

第３図は、ＲＯＭ－ＢＩＯＳのメモリマップを示す図。

第４図は、ＢＩＯＳイメージをロードするための全体的なプロセスを示すフローチャート。

第５図は、マスタブートレコードの形式を示す図。

第６Ａ図は、ＩＢＬルーチンのオペレーションを示すフローチャート。

第６Ｂ図は、ＢＩＯＳイメージを固定ディスクからロードするためのステップを示すフローチャート。

第６Ｃ図は、ＢＩＯＳイメージをディスケットからロードするためのステップを示すフローチャート。

第６Ｄ図は、マスタブートレコードとプレーナ／プロセッサの適合性をチェックする詳細なステップを示すフローチャート。

第７図は、実効可能コードセグメントのオペレーションを示すフローチャート。

第８図は、直接アクセス記憶装置コントローラを示すブロック図。

第９図は、ＩＢＬ媒体を保護するためのディスクコントローラのオペレーションを示すフローチャート。

第１０図は、ＢＩＯＳイメージを保護するための方法を示すフローチャート。

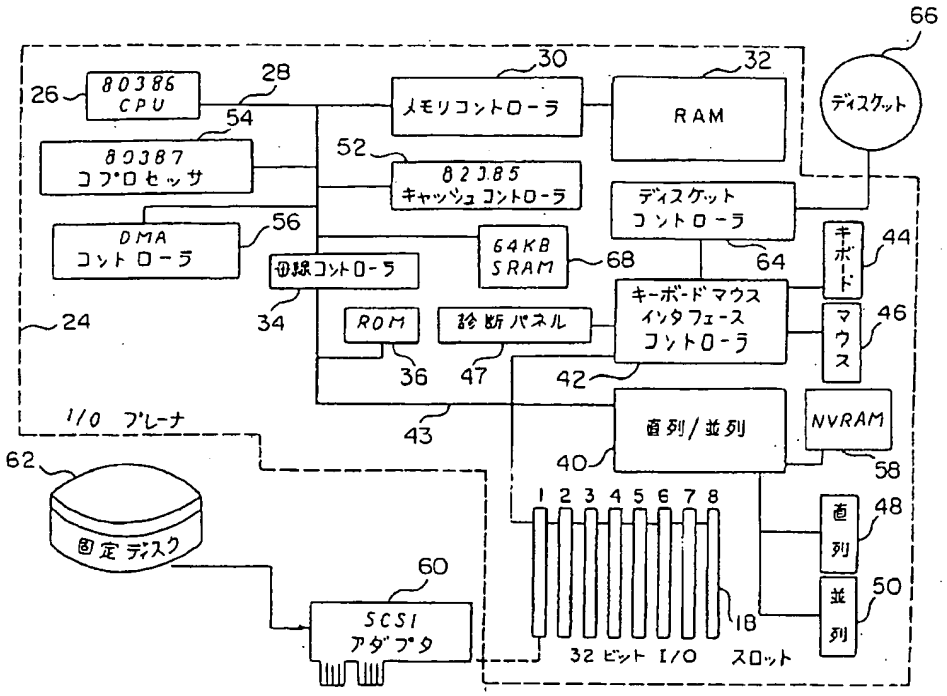


FIG. 1

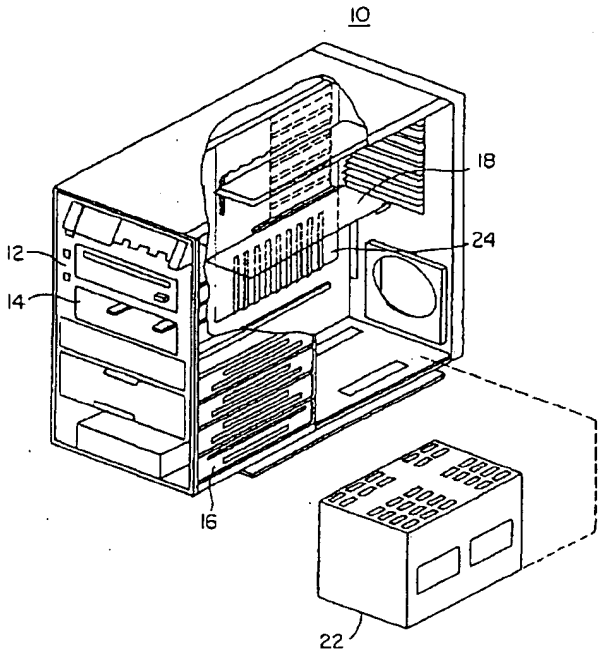


FIG.2

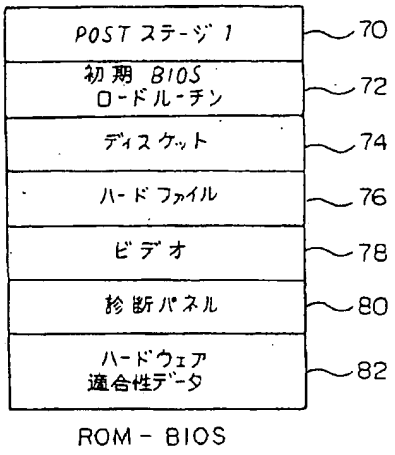


FIG. 3

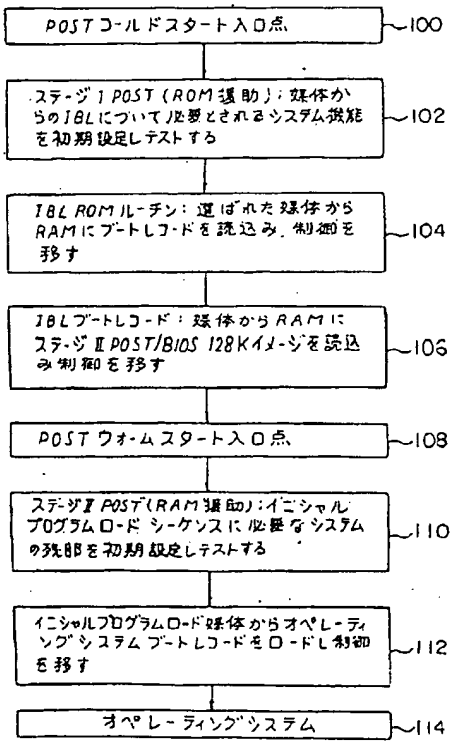


FIG. 4

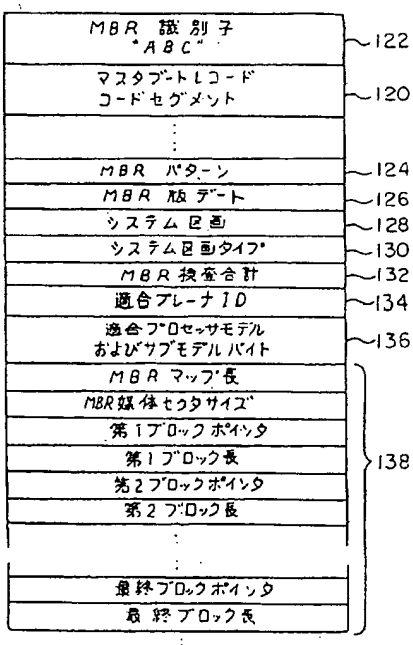


FIG. 5

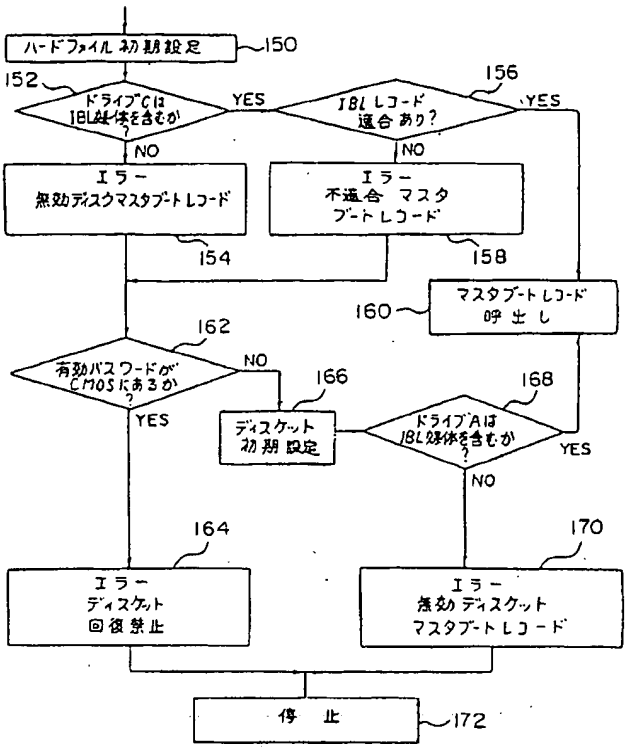

ハードファイル初期設定
150
152
ドライブCは IBL媒体を含むか ?
YES
NO
156
IBL レコード 適合あり?
YES
NO
エラー 無効ディスクマスタブートレコード
154
エラー 不適合 マスタ ブートレコード
158
160
マスタブートレコード 呼出し
162
有効パスワードが CMOSにあるか ?
NO
YES
166
ディスケット 初期設定
168
ドライブAは IBL媒体を含むか ?
YES
NO
164
エラー ディスケット 回復禁止
170
エラー 無効ディスケット マスタブートレコード
停止
172

FIG. 6A

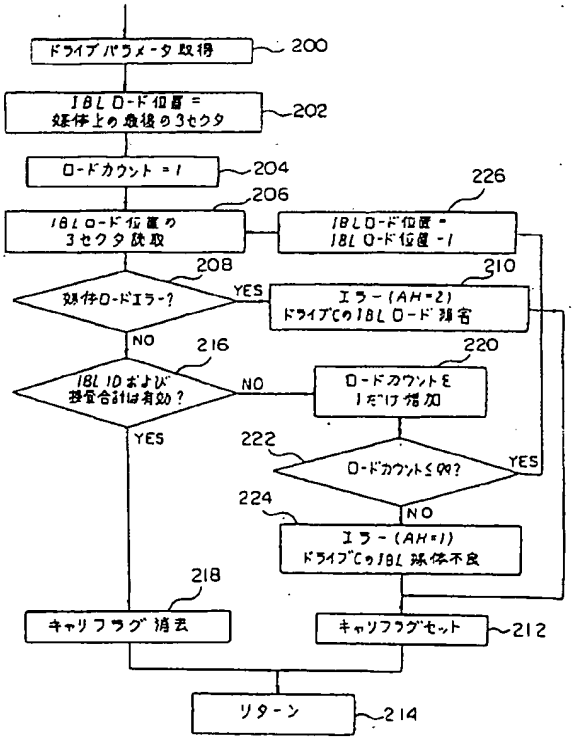

ドライブパラメータ取得
200
IBL ロード位置 = 媒体上の最後の3セクタ
202
ロードカウント = 1
204
206
IBL ロード位置の 3セクタ読取
226
IBL ロード位置 = IBL ロード位置 - 1
208
処理ロードエラー?
YES
210
エラー (AH=2) ドライブCのIBL ロード 異常
NO
216
IBL ID および 容量合計は有効?
NO
220
ロードカウントを 1だけ増加
YES
222
ロードカウント≦99?
YES
NO
224
エラー (AH=1) ドライブCのIBL 媒体不良
218
キャリフラグ消去
キャリフラグセット
212
リターン
214

FIG. 6B

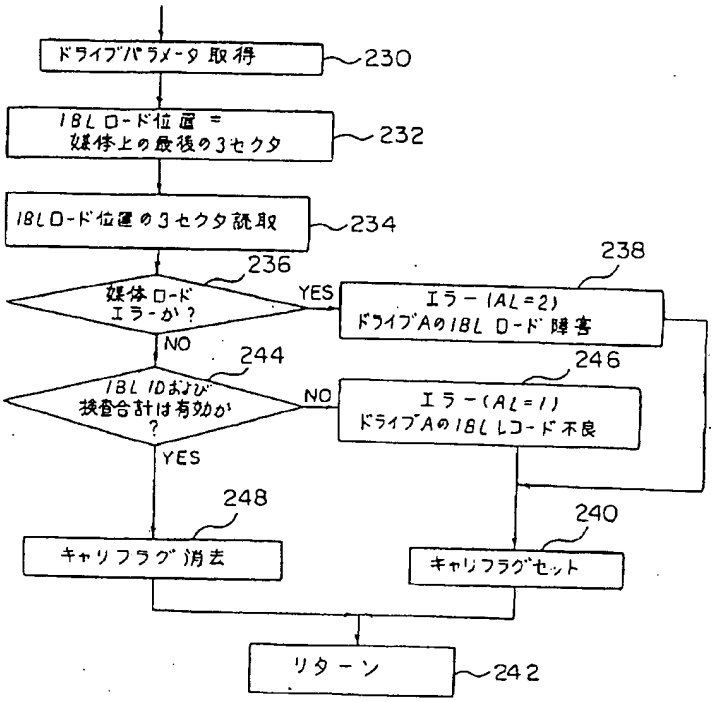

ドライブパラメータ取得
230
IBLロード位置 =
媒体上の最後の3セクタ
232
IBLロード位置の3セクタ読取
234
236
媒体ロード
エラーか？
YES
238
エラー(AL=2)
ドライブAのIBLロード障害
NO
244
IBL IDおよび
検査合計は有効か
？
NO
246
エラー(AL=1)
ドライブAのIBLレコード不良
YES
248
キャリフラグ消去
240
キャリフラグセット
リターン
242

FIG. 6C

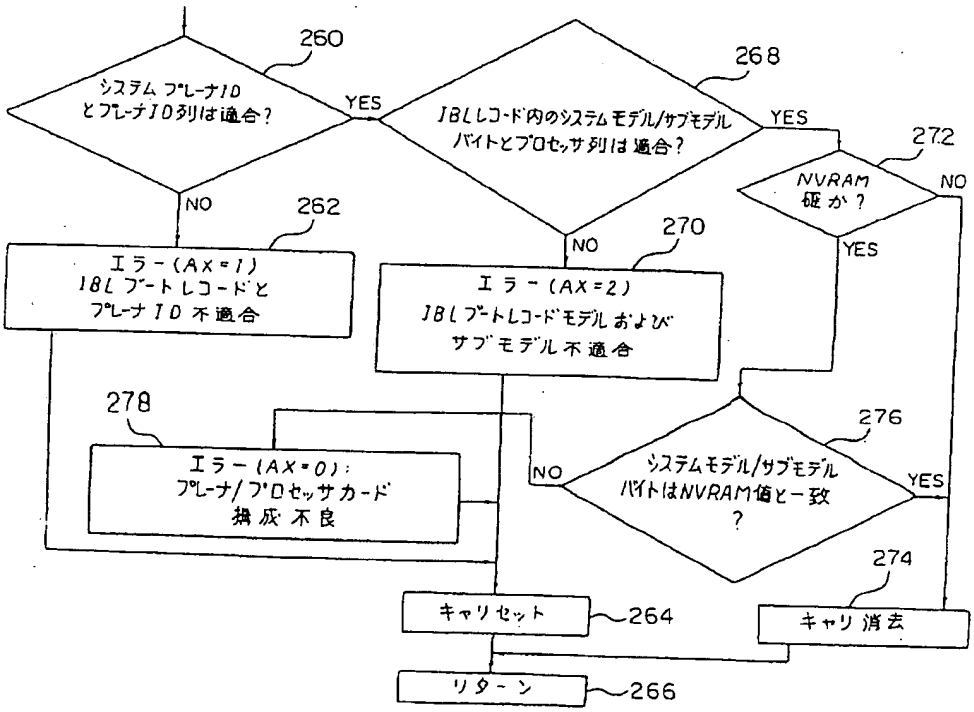

260
システムプレーナID
とプレーナID列は適合？
YES
268
IBLレコード内のシステムモデル/サブモデル
バイトとプロセッサ列は適合？
YES
272
NVRAM
破か？
NO
NO
262
エラー(AX=1)
IBLブートレコードと
プレーナID不適合
NO
270
エラー(AX=2)
IBLブートレコードモデルおよび
サブモデル不適合
YES
278
エラー(AX=0):
プレーナ/プロセッサカード
構成不良
NO
276
システムモデル/サブモデル
バイトはNVRAM値と一致
？
YES
274
キャリセット
264
キャリ消去
リターン
266

FIG. 6D

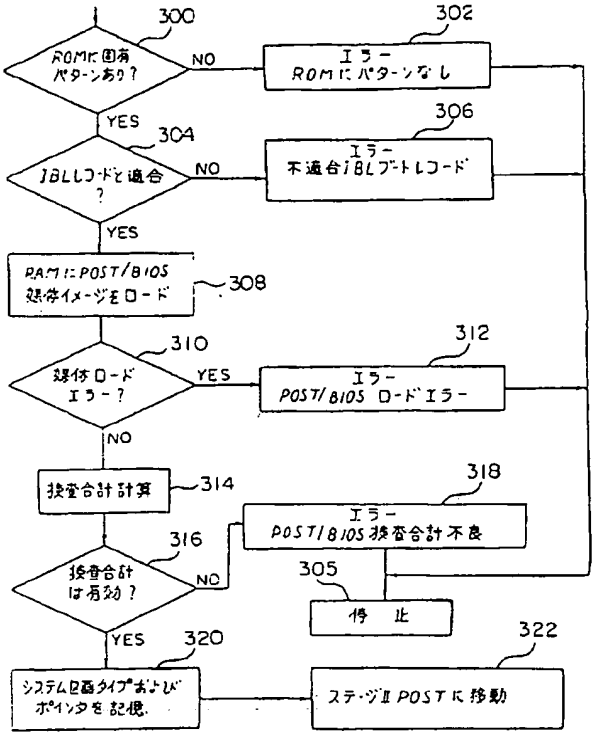


FIG. 7

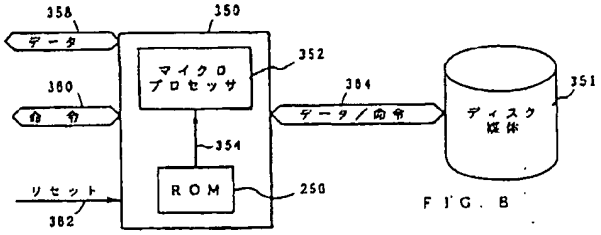


FIG. 8

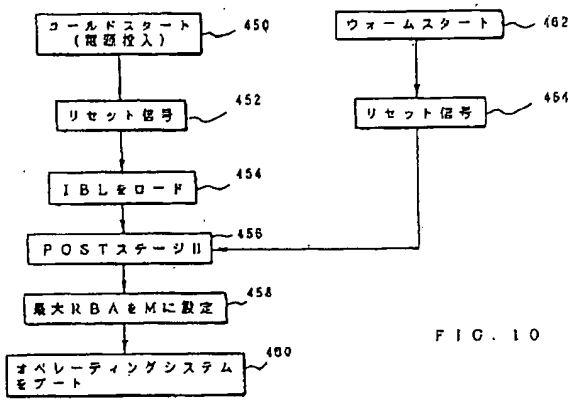


FIG. 10

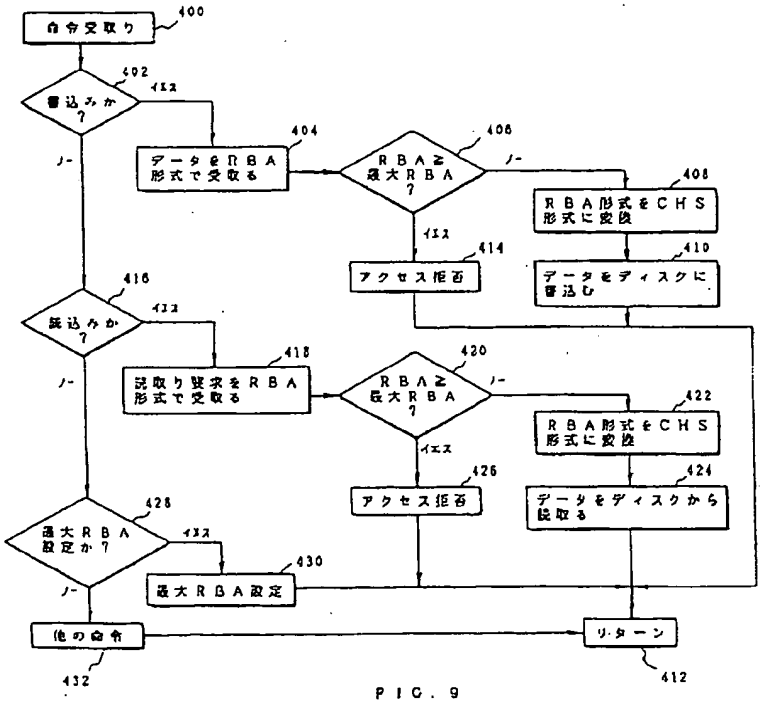


FIG. 9

第1頁の続き

⑤Int. Cl.⁵	識別記号	庁内整理番号
G 06 F 13/10	3 2 0 Z	7218−5B

⑦発 明 者	ドイル・スタンフィール・クロンク	アメリカ合衆国フロリダ州ボカ・ラトン、タウン・ハーバー・ボールヴアード6830地
⑦発 明 者	リチヤード・アレン・ダイアン	アメリカ合衆国フロリダ州ボカ・ラトン、ノース・イースト・73ストリート830番地
⑦発 明 者	スコット・ジエラルド・キニアー	アメリカ合衆国フロリダ州ボカ・ラトン、サドルクリク・ドライブ9005番地
⑦発 明 者	ジヨージ・デイー・コバツク	アメリカ合衆国フロリダ州ボカ・ラトン、ウエストブロック・ドライブ19090番地
⑦発 明 者	マシユー・ステイブン・ボルカ、ジユニア	アメリカ合衆国ノース・カロライナ州ラレイ、ブラス・ケトル・ロード10800番地
⑦発 明 者	ロバート・サクセンマイアー	アメリカ合衆国フロリダ州ボカ・ラトン、ノース・イースト8番コート7329番地
⑦発 明 者	ケビン・マーシヤル・ジボロスキイ	アメリカ合衆国ノース・カロライナ州・ラレイ、ウツドマナー・ドライブ・1313番地
⑦発 明 者	ジエリイ・デユーン・デイキシオン	アメリカ合衆国フロリダ州ボカ・ラトン、エンフイールド・ストリート801番地
⑦発 明 者	アンドリユー・ボイス・マクネイル	アメリカ合衆国フロリダ州デイアフイールド・ビイーチ、ノース・ウエスト41番ウエイ181番地
⑦発 明 者	エドワード・イーブニング・ワツチテル	アメリカ合衆国フロリダ州ボカ・ラトン、ノース・イースト・カイ・テラス500番地